Volume 2, Issue 4, April 2012 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

In-Network Data Management for Wireless Sensor Network

Anuj Kumar Yadav
CSE Department,
National Institute of Technology
Hamirpur (H.P)
anuj.y9@gmail.com

Pratap Singh
CSE Department,
National Institute of Technology
Hamirpur (H.P)

Dr. T.P. Sharma
CSE Department,
National Institute of Technology
Hamirpur (H.P)

Abstract— **Wireless sensor networks have the ability to monitor the physical environment and mainly network used in military and civilians applications. The sensor node has limited energy and there is a key challenge to reduce the energy consumption in the network. The huge amount of data generated in the network to manage the data in the network we propose an index based cooperative caching scheme. These techniques reduce the communication cost for transmitting data and efficiently process the data queries. Index node stores the location of caching nodes so queries are not routed in the whole network.**

Keywords— **Wireless sensor networks, cooperative caching, cache admission control, cache index.**

I. Introduction

A wireless sensor network, consisting of tiny devices which monitor the physical and environmental conditions such as temperature, pressure, motion population etc. At different areas such sensor networks are expected to be widely deployed in vast area of environment for commercial, civil, military applications such as surveillance, vehicle tracking, climate and habitat monitoring, intelligence, medical and acoustic data gathering. The key limitation of wireless sensor networks is storage power and data processing. Radio communication is the major source of energy consumption in WSN and hence in recent years, various schemes have been proposed to reduce the internode transmissions. This is achieved either by reducing the amount of data to be disseminated from source to sink or by reducing the path traversed by queries and data. Strategies like data compression (Natto and Sharma, 2005), in-network data processing (Faso et al., 2007), source data filtering (Kadayif and Kandemir, 2004), better topology management, caching and indexing reduce the length of path traversed by data and/or queries in the network.

In past many data dissemination scheme proposed like external storage, local storage, data centric storage-based in (DCS) event to be detected by name and sensing data of these event are stored at the node with in the network instead of being external storage(Ratnasamy et al.2003). Unnecessary transfer of sensing data can be avoided in this storage scheme using direct diffusion and the two-tier data dissemination scheme, in which source send the data to the sink when sink has queried the data. In TTDD, the source detecting the certain event and broadcast event in the whole network, and sink interested in the event can send their queries directly to the source.

In this paper, we propose index based cooperative caching scheme. In this scheme, as shown in fig. 1 sink issue a query to the index node and index indicate the location information of the data. When sink issue a query it first go to the index cache in index cache there is a location of storing cache and index node send the query to the storing cache. Storing cache send the data direct to the sink. This process reduced the query path. To make an index in the network we use a geographic hash table in this we use a hash function H (.) which has a unique key to each storage.

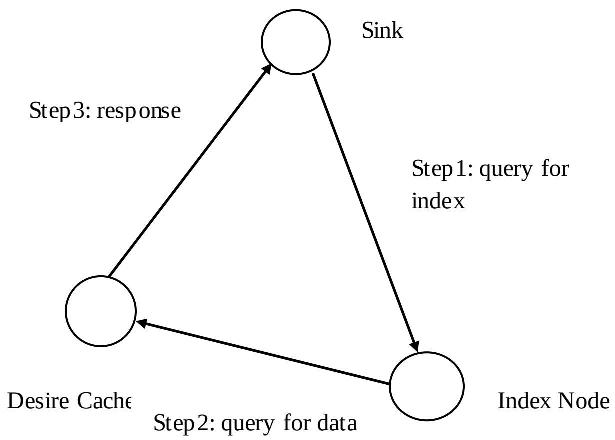

Fig. 1 Index based cooperative caching scheme

II. Related Work

Ratnasamy et al. proposed GHT: A geographic hash table for data centric storage. In GHT hashes key in into geographic coordinates, and store a key-value pair at the sensor node geographically nearest the hash of its key. The systems replicate store data locally to ensure persistence when node fails. It uses an efficient consistency protocol to ensure that key-value pairs are stored at the appropriate nodes after topological changes. In these systems, a data object is associated with a key and each node in the system is responsible for storing a certain range of keys. A name-based routing algorithm allows any node in the system to locate the storage node for an arbitrary key. This enables nodes to put and get files based on their key the core step in GHT is the hashing of a key k into geographic coordinates. Both a Put () operation and a Get () operation on the same key k hash k to the same location.

Zhang et al. proposed data dissemination with ring-based index for wireless sensor networks this proposed scheme reduced the communication cost for transmitting the data and to efficiently process to data queries. This scheme based on the idea that sensing data are stored at the nodes closed to the detecting nodes and the location information of these storing nodes is pushed to some index nodes closed to them. Hence, queries for a particular event are directly routed to the appropriate index nodes, which may then forward the queries to appropriate storing nodes. This scheme avoids both unnecessarily transferring the sensing data and flooding control messages throughout the network. A hash function H (.) can be constructed which maps each target type to one location within the field.

Ratanasamy et al. proposed a data centric storage scheme in this approach, relevant data is stored, by name, at nodes within the sensornet. Queries for data with a particular name can then be sent directly to the node storing that named data, thereby avoiding the flooding required in some data-centric routing proposals. The mechanism of data centric storage is that Data-centric storage is basically a distributed hash-table, supporting two basic primitives:

1. Put (key, value) this stores value (the data) according to the key (Which is based on the name of the data)
2. Get (key) this retrieves whatever value is stored at the key, (Which is derived from the event name?)

Given an event name, we hash that name into a key. This key is allocation somewhere in the sensornet. Thus, the hash function must be chosen with knowledge of the boundaries of the sensornet. The put (key, value) command sends a packet with the given pay load into the sensornet routed towards the location key. Our (slightly) modified form of GPSR will forward the packet to the sensornet node closest to this location, where the data can be stored; we will call this closest node the home node for an event. Similarly a get (key) command is forwarded to the sensornet closest to the key location, and that node then returns a packet to the source of the query with the corresponding data.

T.P. Sharma et. al proposed an effective caching strategy based on the cooperation among various sensor nodes to form a large cumulative cache. In this strategy node along with its own storage utilizes storages of other nodes in a defined zone around it to realize a large cumulative cache. A novel token based cache admission control is devised where node holding cache token can only cache or replace data items. A cache discovery mechanism to efficiently fetch copy of data item. An item replacement policy is developed based on least utility means; a data item with least utility is removed from cache.

III. COOPERATIVE CACHING

Low power radio mode of a sensor node is exploited to from a cooperative region. In our approach, a circular region of radius RL/2 around each DN including sink and SDN is defined as Cache Zone (CAZ). Let SCS is the set of nodes in CAZ. Nodes in SCS cooperate among themselves and with DN by sharing their local caches to realize much larger cumulative cache (CMC). All nodes in SCS communicate with each other and with DN using low power radio for every cache management activity. The reason for taking radius of CAZ as RL/2 is twofold; first all cache nodes in CAZ should communicate in single hop with each other and second to utilize low power radio for caching so as to conserve energy. Each DN probes its CAZ by broadcasting special signal in it and each sensor node in SCS respond with a tuple comprising its id (geographical coordinates) and residual energy. This way, DN becomes aware of nodes available for caching and prepares an indexed list of node-ids in descending order of residue energies. CAZ formation and path setup mechanism yield an appropriate model for data dissemination using cooperative caching.

IV. CACHE ADMISSION CONTROL

To know where we cache the data item in the network first we calculate the Weight Function (W)

- W (Level, Energy, Degree, Id)
- Level= hop distance from the root
- Energy= residual energy to total energy
- Degree= number of neighbor of a DN at level-l
- Id= node unique id

V. ALGORITHM

1. Set a path r∈v(using aggregation technique)
2. Compute the hop distance from r to each node(using breadth first search)
3. Every node exchange information with its nodes
4. Color every node in v white
5. While a white node in v exits do
6. The white node u send a message to s hop neighbor v about w
7. V becomes a green and broadcast a green message to its s-hop neighbors
8. If receive a green message, white node become a gray.

Algorithm 1 shows the cache admission control strategy. In this first we set the path between sink and SDN using aggregation technique then root initializes the BFS (to compute the hop distance). Every node exchanges information with its s-hop neighbors about the level, energy, degree, id. All nodes should know all its s-hop neighbors as a result.

Secondly the sink child U calculates the W (L, E, D, Id) and sends a message its exact s-hop away node. When V receives this message, it become green and broadcast a green message to all of its s-hop neighbors. Upon receiving green message child nodes become gray and broad cast gray message to all of its s-hop neighbors. This procedure stops when all nodes information reaches to the SDN.

Source Dissemination Nodes know all nodes W function. When the sink issue a query, SDN sends query to sink and store this query in the network according to the size of query. The SDN store this query with the time stamp and this store information send to the index node.

V. CACHE DISCOVERY

To discover a desire cache node we make an index cache node closer to sink. This node cache all information about the which node to cache the data item. Before sending query for data item sink first check in index node to the location of the data item. A hash function H (.) can be constructed which map each target type to one location with in the field. We make an index node is I-hop away from the sink. We cache the data item in the network with name and time stamp this information is hash and send to the index node.

Hash value $\rightarrow$ key length + associated value of keys first characters + associated value of keys last character

Two functions is used to store and retrieve information for store we use put function-put (key, value) and retrieve data we use get (key).

Example $\rightarrow$ elephant
a=9, b=3, d=5, e=1, l=4, p=0, h=6, n=8, t=2
Key length=8
Associated value of keys first characters=1
Associated value of keys last characters=2
Hash value $\rightarrow$ key length + associated value of keys first characters + associated value of keys last character
Hash value= 8+1+2= 11
In index node we store this value + time stamp (at the time query generated) and location of this value.

When sink issue a query in index node it first check the time stamp if time stamp is miss it send the query direct to the SDN. If hits the time stamp, then see the value and location of the data item where it stored.

VII. CONCLUSIONS

We propose index based cache scheme to support the manage the data in the wireless sensor network. This scheme is based on the idea that sensing data are collected, processed and stored at the nodes close to the detecting nodes and location information of these storing nodes is pushed to some index nodes. This scheme can extend the battery life of node and reduced the communication cost in the network and provide minimizing the use of limited network and computational resources.

ACKNOWLEDGMENT

We express our thanks to Dr. T. P. Sharma, the Associate, Prof. NIT Hamirpur, and my Thesis Supervisor, for the encouraging environment provided.

REFERENCES

[1] Ratnasmay, B. Karp, S. Shenker, R. Govidan, "Data-Centric Storage in Sensornets with GHT, a geographic hash table. Mobile Networks Applicant", 8 427-428.
[2] Zhang, W. S., G. H. Cao and T. La Porta, 2007. Data dissemination with ring-based index for wireless sensor networks. IEEE Trans. Mobile Comput., 6: 832-847.
[3] ChaoGao, Bingwen Wang, Xiaoya Hu, Zhuguo Li and Hongliang Gao "An Efficient Index-based Data Storage Method for Wireless Sensor Networks " Information Technology Journal10(10) 1934-1941.
[4] S. Shenker, S. Ratnasamy, B. Karp, R. Govindan, and D. Estrin, "Data-centric storage in sensornets", Under submission to ACM SIGCOMM HotNets, Jul. 2002.
[5] Thang Nam Le, Wei Yu, Xiaole Bai and Dong Xuan "A Dynamic Geographic Hash Table for Data-Centric Storage in Sensor Networks" IEEE Communication Society WCNC 2006.
[6] T.P. Sharma, R.C. Joshi, Manoj Misra, "Cooperative Caching for Homogeneous Wireless Sensor Networks" Int. J. Communication Networks and Distributed Systems, Vol.2, No. 4, 2009.
[7] R.J. Chihelli, "Minimal Hash Functions Made Simple", *Published in communications of ACM*, Vol. 23, No 1,pp. 17-19.
[8] Fasolo,E. Rossi, M. Widmer, J. Zori, M.(2007) 'In-network aggregation techniques for wireless sensor networks: a survey' ,IEEE Wireless Communications, vol. 14, issue2, pp. 70-78, April.
[9] Kadayif, I. Kandemir, M. (2004) 'Tuning in-sensor data filtering to reduce energy consumption in wireless sensor networks', Proceedings Of the Design, Automation and Test in Europe Conference and Exhibition (DATE'04), vol.2, pp. 852-857, 16-20 February.
[10] Naoto, Kimura. Shahram, Latifi. (2005) 'A Survey on Data Compression in Wireless Sensor Networks', Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'05), vol. 2, pp. 8-13, April.